

```
3 print('@ builderlib module start')
4
5 class Builder:
6     print('@ Builder body')
7
8     def __init_subclass__(cls) -> None:
9         print(f'@ Builder.__init_subclass__({cls!r})')
10
11         def inner_0(self):
12             print(f'@ SuperA.__init_subclass__:inner_0({self!r})')
13
14         cls.method_a = inner_0
15
16     def __init__(self) -> None:
17         super().__init__()
18         print(f'@ Builder.__init__({self!r})')
```

```
20 def deco(cls):
21     print(f'@ deco({cls!r})')
22
23     def inner_1(self):
24         print(f'@ deco:inner_1({self!r})')
25
26     cls.method_b = inner_1
27     return cls
```

```
31 class Descriptor:
32     print('@ Descriptor body')
33
34     def __init__(self) -> None:
35         print(f'@ Descriptor.__init__({self!r})')
36
37     def __set_name__(self, owner, name):
38         args = (self, owner, name)
39         print(f'@ Descriptor.__set_name__({args!r})')
40
41     def __set__(self, instance, value):
42         args = (self, instance, value)
43         print(f'@ Descriptor.__set__({args!r})')
44
45     def __repr__(self) -> str:
46         return '<Descriptor instance>'
47
48 print('@ builderlib module end')
```

```
3 from builderlib import Builder, deco, Descriptor
4
5 print('# evaldemo module start')
6
7 You, 3 days ago | 1 author (You)
8 @deco # <1>
9 class Klass(Builder): # <2>
10     print('# Klass body')
11
12     attr = Descriptor() # <3>
13
14     def __init__(self):
15         super().__init__()
16         print(f'# Klass.__init__({self!r})')
17
18     def __repr__(self):
19         return '<Klass instance>'
20
21 def main(): # <4>
22     obj = Klass()
23     obj.method_a()
24     obj.method_b()
25     obj.attr = 999
26
27 if __name__ == '__main__':
28     main()
29
30 print('# evaldemo module end')
```

```
>>> import evaldemo
@ builderlib module start # 1
@ Builder body
@ Descriptor body
@ builderlib module end
# evaldemo module start
# Klass body # 2
@ Descriptor.__init__(<Descriptor instance>) # 3
@ Descriptor.__set_name__(<Descriptor instance>,
    <class 'evaldemo.Klass'>, 'attr') # 4
@ Builder.__init_subclass__(<class 'evaldemo.Klass'>) # 5
@ deco(<class 'evaldemo.Klass'>) # 6
```